

5

Remarks

In the Office Action dated July 30, 2003 (hereinafter "Office Action"),
Claims 1-2, 5-6, 11-14 and 19-20 were rejected, and Claims 3, 4, 7-10, 15-18,
21, and 22 were indicated as including allowable subject matter. The Claims
remain pending as previously presented. In view of the arguments presented
10 below, it is respectfully submitted that all Claims are in condition for allowance.

1. Applicants' Representative appreciatively acknowledges the withdrawal of
the previous Final Rejection.

15 2. Claims 1, 2, 5, 11-12, and 19-20 were rejected under 35 USC §103(a) as
being unpatentable over U.S. Patent No. 6,029,006 to Alexander et al.
(hereinafter "Alexander"). This rejection is respectfully traversed.

Applicants' Claim 1 describes a data processing system having an
instruction pipeline capable of initiating simultaneous execution on a variable
20 number of instructions in a predetermined period of time. The system includes a
first storage device to store a programmable count value, and a logic sequencer to
generate a pipeline control signal to cause the instruction pipeline to initiate
concurrent execution on a predetermined number of instructions indicated by the
count value within the predetermined period of time. In one embodiment, the
25 predetermined period of time is determined by the number of stages in the pipeline.

The Examiner states that Applicants' invention of Claim 1 is taught by
Alexander. In particular, Alexander's use of the interval register is said to teach
Applicants' use of a count value. Applicant's Representative respectfully
disagrees with this assertion.

30 The Alexander interval field "...is used to specify the number of machine
cycles required between successive fetches of instructions from instruction cache
14." (Alexander column 12 lines 60-63.) In other words, the Alexander interval
value is used to determine the delay between the entry into the pipeline of two
successive instructions in the instruction stream. Therefore, if the Alexander
35 interval is set to "one", it would appear that three successive instructions X, Y,

5 and Z will enter the Alexander pipeline with a one-cycle delay between instructions X and Y, and another cycle delay between instructions Y and Z, as follows:

Stage:		1	2	3	4	5
Row:						
10	Cycle 1	X				
	Cycle 2		X			
	Cycle 3	Y		X		
	Cycle 4		Y		X	
	Cycle 5	Z		Y		X

15

Thus, the Alexander interval value most certainly does not control how many instructions begin concurrent execution within the pipeline in any predetermined period of time. For example, three instructions begin concurrent execution within the pipeline illustrated above during a predetermined time period equal to
20 five cycles. This number of instructions is unrelated to the interval value of "one".

In contrast to the Alexander system, Applicants' count value precisely controls how many instructions begin execution during a predetermined period of time, which in one embodiment, is equal to six clock cycles. For instance, when Applicants' count value is set to one, exactly one instruction enters Applicants' pipeline during six clock cycles, as shown in Applicants' Figure 10. This is very
25 dissimilar to the effects obtained when the Alexander interval value is set to "one", as can be seen above.

As can be appreciated by the foregoing discussion, the pipeline control mechanism described in Alexander is quite different from that included within Applicants' system. For example, when Applicants' count value is set to "four", exactly four instructions enter the pipeline and begin concurrent execution, as shown in Figure 13. Similarly, when Applicants' count value is set to "five", exactly five instructions enter the pipeline and begin concurrent execution during six clock cycles, as shown by Applicants' Figure 14, and so on. This precise
30 control is provided by a complex set of control sequences described within
35

5 Applicants' Specification and illustrated within the various timing and logic
diagrams. This complex mechanism allows Applicants' system to "de-pipe", or
partially clear, in the most efficient manner possible to handle specific timing
conflicts. Similar control capabilities are not available within the Alexander
system. For instance, in the Alexander system, there is no interval value that
10 may be selected that will allow exactly four instructions to enter the Alexander
pipeline and begin concurrent execution during a predetermined period of time.
Similarly, there is no interval value that will allow precisely five instructions to
enter the pipeline during a predetermined period of time, and so on. This loss of
precision does not matter in the Alexander system since that system is designed
15 to slow processor throughput by an inexact amount necessary to reduce heat
generation. However, this precise control is useful in Applicants' system, which
is designed to resolve pipeline conflict situations.

Prior art systems similar to that disclosed in Alexander are discussed in
Applicants' Specification in regards to the foregoing limitations, as follows:

20

"A system for controlling pipeline execution in a programmable
manner is described in U.S. Patent No. 5,911,083 entitled
"Programmable Processor Execution Rate Controller" to Kuslak,
which is assigned to the assignee of the current invention. This
25 patent describes a system *for preventing additional instructions
from entering the instruction pipeline for a selected amount of time*
after selected ones of the instructions enter the instruction pipeline.
This may be referred to as "de-piping the pipeline.... [This prior art]
de-piping mechanism can not be used to efficiently solve timing
30 conflicts that are caused by two non-contiguous instructions within
the instruction stream, or that are caused by a combination of more
than two instructions. This is because the *prior art de-piping
mechanism inserts delay into the pipeline immediately following a
particular instruction* instead of controlling the number of
35 *instructions that are concurrently executing within the pipeline.*"
(Applicants' Specification page 5 lines 4-11 and 24-28, emphasis
added.)

The above-mentioned problems associated with prior art designs such as the
40 Alexander system are solved by Applicants' system in a manner described
throughout Applicants' Specification.

5 To summarize, Applicants' Specification discusses the limitations associated with prior art systems such as described in Alexander, which inserts predetermined delay between successive instructions in the pipeline in a way that does not provide the capabilities offered by Applicants' system. Claim 1 is allowable over Alexander for at least the following reason:

10 a.) Alexander does not teach or suggest a device to store a programmable value indicative of a predetermined number of instructions that will enter a pipeline to begin concurrent execution. Instead, the Alexander interval value specifies the number of machine cycles that elapse between the entry into the pipeline of two successive instructions within the instruction stream.

15 b.) Alexander does not teach or suggest a logic sequencer to cause the instruction pipeline to initiate concurrent execution on the predetermined number of instructions in a predetermined period of time.

Next, the Official Notice taken by the Examiner is discussed. The
20 Examiner takes Official Notice that it was known at the time of the invention to implement synchronous pipelines. The Examiner further states that using this type of implementation, one of ordinary skill in the art would understand that the processor defines a predetermined period of time, and the interval of Alexander defines the variable number of instructions that are being executed in the pipeline
25 by allowing only so many instructions in the defined period of time of the pipeline. (Office Action page 3, last sentence.)

The Examiner's statements will be addressed in turn. First, it will be assumed for discussion purposes that synchronous pipelines were known in the art at the time the invention was made. Synchronous pipelines generally
30 receive some type of clock signal. The clock cycle, or a multiple thereof, might be said to define some "predetermined period of time". However, this in no way teaches or suggests Applicants' mechanism that allows a user to programmably specify the number of instructions that will begin concurrent execution within a pipeline in a predetermined period of time. Moreover, while it is true that the
35 Alexander interval value will limit the number of instructions entering the pipeline

5 by inserting delay between successive instructions, that interval value does not in
any way teach or suggest use of a programmable count value to control the
predetermined number of instructions that will begin concurrent execution within
the pipeline in a predetermined period of time. For example, unlike Applicants'
invention, there is no interval value that can be selected in Alexander to control
10 the pipeline so exactly four or five instructions will enter the pipeline to begin
concurrent execution within the predetermined period of time. Moreover, as
illustrated above, when the interval value is "one", the Alexander pipeline is not
controlled to allow only a single instruction to enter the pipeline. The same can
be said for any other interval value that may be selected.

15 For all of the foregoing reasons, Alexander does not teach or suggest
Applicants' pipeline control mechanism of Claim 1, and this rejection should be
withdrawn.

Claims 2 and 5 depend from Claim 1, and are allowable over this rejection
20 for at least the reasons discussed above in reference to Claim 1.

Independent method Claim 11 includes aspects of Applicants' invention
that are similar to those discussed above in reference to Claim 1. For reasons
similar to those discussed above with respect to Claim 1, Claim 11 is allowable
25 over this rejection, which should be withdrawn.

Claim 12 depends from Claim 11, and is allowable over this rejection for at
least the reasons discussed above in reference to Claim 11.

Independent apparatus Claim 19 includes aspects similar to those
30 discussed above in reference to Claim 1. For reasons similar to those discussed
above in reference to Claim 1, Claim 19 is allowable over this rejection, which
should be withdrawn.

Claim 20 depends from Claim 19, and is allowable over this rejection for at
least the reasons discussed above in reference to Claim 19.

35

5 For the foregoing reasons, it is respectfully submitted that Claims 1, 2, 5, 11-12, and 19-20 are allowable over this rejection, which should be withdrawn.

3. Claims 6 and 13-14 were rejected under 35 USC §103(a) as being unpatentable over Alexander et al. in view of U.S. Patent No. 5,996,064 to Zaidi et al. (hereinafter, "Zaidi") and further in view of U.S. Patent No. 6,345,362 to Bertin et al. (hereinafter "Bertin"). This rejection is respectfully traversed.

Claim 6 depends indirectly from Claim 1 and is allowable for the reasons set forth above with respect to Claim 1. Claim 6 further includes the aspect of Applicants' invention wherein the detection of a predetermined instruction can cause the de-piping of Applicants' pipeline to a level determined by a count value respectively associated with that instruction. (See Applicants' Specification page 29 line 19 through page 30 line 2 in reference to Figures 8A, 8B, and 9.) This capability allows certain known timing conflicts to be resolved in a very efficient manner. The Examiner states that this aspect of Applicants' invention is taught by Zaidi.

Zaidi describes a system for scheduling instructions within an out-of-order processor. The system inserts a "post-ready latency" delay between the execution of an identified instruction and the preceding instruction. The specific delay to be inserted is associated with the identified instruction using a table stored within memory. This ensures that proper operation occurs within the processor. Thus, Zaidi, like Alexander, teaches a system for inserting delay between successive instructions. Zaidi adds nothing to Alexander to teach or suggest utilizing a specific count value to control the number of instructions executing concurrently within the pipeline.

30 In addition to the foregoing, there is no motivation to add the Zaidi post-latency values to the Alexander system. Alexander discloses a system related to throttling instruction execution to reduce power consumption. The Alexander system throttles instruction execution based on temperature measurements. Nothing in Alexander suggests that there would be any reason to recognize individual instructions. Moreover, one skilled in the art would not be inclined to

35

5 look to the Zaidi system, which describes a mechanism for controlling the
execution of out-of-order instructions within a high-speed processor, for any
teaching related to limiting power consumption within a processor.

Next, the Examiner's assertions regarding Bertin are considered. The
Examiner states that Bertin demonstrated that it was known to attempt to control
10 instructions based upon power usage. (Office Action page 5, last paragraph.)
Thus, the Examiner appears to be asserting that the motivation to combine
aspects of Zaidi with the Alexander system may be found in Bertin. Applicants'
Representative respectfully disagrees. Bertin discloses a system to control the
power consumption of various functional units, as may be useful within a portable
15 electronic device. Functional units are in a high power state when instructions
requiring their use are to be executed. Conversely, functional units are in a low
power state when they are not involved in current instruction execution. (Bertin
column 2 lines 30-33.) An execution unit allows execution of an instruction at
current processor speeds if the functional units that are required to execute the
20 instruction are operating at the required power levels. Otherwise, the execution
unit may stall the instruction stream for a time sufficient to allow voltage levels to
ramp up to the required levels. (Bertin column 3 lines 3-15.)

The Bertin system appears to be particularly adapted for use with battery-
powered portable integrated circuits. This type of system is not adapted for use
25 with a high-speed instruction pipeline. Specifically, the Bertin process of
periodically powering on, then off, various functional units would not be feasible
in modern high-speed processor pipelines that typically measure instruction
cycles in nanoseconds or picoseconds. Such operations would decrease
processing throughput beyond what would be practical. For this reason, one
30 skilled in the art would not look to Bertin for any teaching concerning high-speed
pipeline technology, and would not be motivated by Bertin to include any aspects
of Bertin or Zaidi with Alexander.

In addition to the foregoing, the Bertin system is based upon identifying
instructions so that relevant functional units can be powered on and off as
35 required by the identified instructions. The Examiner asserts that such

5 functionality would be usefully incorporated into Alexander to identify energy
hungry instructions. (Office Action page 6, first paragraph.) This is not
understood. Modern pipelined instruction processors execute millions of
instructions per second. Within this context, it is unclear how or why one would
attempt to control the Alexander instruction throttling based on the identification
10 of individual instructions. No one instruction will provide any useful indication of
power consumption levels. Instead, such an indication is provided by the
processor activities occurring over a relatively large period of time, such as one
or more seconds. During that time, millions, or even tens of millions, of
instructions, are executed. Thus, rather than throttling processor throughput
15 using individual instruction identification, a workable approach employs
temperature as a gage for average processor activity.

For the all of the above reasons, one skilled in the art would not be
motivated to combine any aspects of Bertin or Zaidi with Alexander. In sighting
these unrelated teachings, the Examiner is attempting to piece together
20 Applicants' invention in hindsight, something that has long been held
impermissible. Thus, Claim 6 is allowable over this rejection, which should be
withdrawn.

Claim 13 depends indirectly from Claim 11 and is allowable for the
25 reasons set forth above with respect to Claim 11. Claim 13 further includes the
aspect of Applicants' invention that are similar to those discussed above in
reference to Claim 6. For the additional reasons discussed above in reference to
Claim 6, Claim 13 is allowable over this rejection, which should be withdrawn.

Claim 14 depends from Claim 13 and is allowable for at least the reasons
30 discussed above in reference to Claim 13.

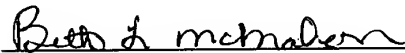
4. Applicants' Representative appreciatively acknowledges the indication of
allowable subject matter in Claims 3-4, 7-10, 15-18, and 21-22. It is respectfully
submitted that in view of the arguments presented above, these Claims are
35 allowable as presently presented.

5

Conclusion

In the Office Action dated July 30, 2003 (hereinafter "Office Action"),
Claims 1-2, 5-6, 11-14 and 19-20 were rejected, and Claims 3, 4, 7-10, 15-18,
21, and 22 are indicated as including allowable subject matter. The Claims
remain pending as previously presented. In view of the arguments presented
10 above, it is respectfully submitted that all Claims are in condition for allowance,
and an early Notice of Allowance is respectfully requested. If the Examiner has
any questions or concerns, a call to the undersigned is encouraged and
appreciated.

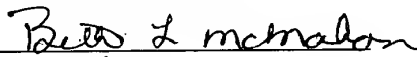
Respectfully submitted,



Beth L. McMahon
Reg. No.: 41,987
Tel. No.: (651) 635-7893
Unisys Corporation
M.S. 4773
P.O. Box 64942
St. Paul, MN 55164-0942

I hereby certify that this correspondence is being deposited in the United States Postal Service as first
class mail in an envelope addressed to: Mail Stop Non-Fee Amendment, Commissioner for Patents, P.O.
Box 1450, Alexandria, VA 22313-1450 on October 29, 2003.

Beth L. McMahon
Attorney for Applicants


Signature

10/29/03
Date of Signature